OIPE JC67
NOV 24 2003
PATENT & TRADEMARK OFFICE

Express Mail No:
EV336614202US

RECEIVED
DEC 0 2 2003
Technology Center 2100

Translation of Appendix A

PATENT PROPOSAL

A METHOD AND STRUCTURE FOR OPTIMIZING CODING AND COMPUTING FUZZY INFERENCE

Authors: Pappalardo Francesco, Giacalone Biagio, Palano Marcello

Foreword

A detailed discussion of the prior art concerning fuzzy theory is omitted as this is repeatedly indicated in our earlier patents and we focus only on those fuzzy concepts that are essential for describing the instant patent proposal.

We recall that in fuzzy processing one has:

- an input variable,
- a fuzzy set of the input variable,
- membership functions contained in the fuzzy sets,
- logical operators (AND and OR) of the fuzzy type, and
- a consequent.

The fuzzy inference or fuzzy rule used in fuzzy computation is, according to fuzzy theory, of the type:

IF antecedent THEN consequent

where the antecedent part can generally be expanded into an expression of the type

ing0 is/not_is MF0 and/or ing1 is/not_is MF1... and/or ingn is/not_is MF_n.

Therefore, the generic fuzzy rule, such as that shown above, consists of an antecedent made up of atomic conditions (such as "**ing0 is/not_is MF0**", which can be denoted for the sake of brevity simply as "**V is/not_is**

M") related in logical connection by operators such as AND, OR.

The atomic condition expresses the degree to which one element of the discourse universe has membership of a particular fuzzy subset of this universe. The element in question is denoted by the input variable V and the fuzzy subset is characterized by the membership function M.

Additionally, the fuzzy rule is encoded and stored within the structure which is to compute it.

All the methods of storing the fuzzy inference and the corresponding knowledge base associate each input (in a way similar to the procedure used in fuzzy theory) with the set of which it is a member, in other words the corresponding knowledge base which contains all the memberships used by the input variable.

For calculating the fuzzy inference by means of a structure capable of computing it, the discourse universe of the membership functions of all the input variables is translated into a base discourse universe in such a way that the fuzzy inference can be computed. In the case of a calculation structure of the numerical type, the base discourse universe for all the membership functions will be mapped on to a discrete set which extends from 0 to 2^n-1, where n is the number of bits specified as the size for all the input variables.

Problems encountered one wishes to optimize

1) The fields of the discourse universe of the various input variables are virtually always completely different from each other: consequently, the memberships are also formally different; furthermore, even if they have the same discourse universe, it is not easy to

recognize two identical membership functions described in a graphic way.

Moreover, for the various input variables, there are frequently memberships which are identical to each other in the base discourse universe; consequently, in the storage within the structure which is to compute them, these memberships are repeated for each input variable with which they are associated.

What has been stated in the foregoing, that is the problem of redundancy of information due to memorizing the same membership function for different variables or the same variable, has not been optimized so far.

2) The various atomic conditions of the type "**ing0 is/not_is MF0**", suitably calculated, yield values, called **alpha** (which become the operands of the antecedent part). The atomic conditions are calculated as points of intersection between the input variable **ing0** and the membership function **MF0**; this calculation is a time-consuming operation, and occupies a considerable part of the calculation time in the total calculation of the inference.

Basic idea underlying the patent proposal

As stated in the foregoing, the fields of the discourse universe of the various input variables are virtually always completely different from each other, and that the memberships suitable for the type of fuzzy operation to be executed have to be described for each variable in the field in which it exists. In order to be used by the calculation unit which computes the inferences, these memberships have to be remapped from the discourse universe of the variable in a discretized interval suitable for the calculation unit. This operation has to be executed for all the discourse

universes of the input variables used in the inferences to be computed.

It has been noted that, after the remapping of the discourse universe, many membership functions are found to be identical, both among those of the same input variable and, very frequently, among those of different input variables.

Moreover, the value of the variable V (which represents an input of the system) used for calculating the operands of the antecedent part (alpha values) changes, in many cases, at a very low frequency, and therefore the result is that the same value of alpha is calculated cyclically many times.

The patent proposal indicated in the following makes it possible:

1) to optimize the storage space used, eliminating redundancy, and

2) to optimize the inference execution time, by storing the previously calculated alpha values in a back-up store.

The circuit considered is shown in figure 1.

In the circuit it is worth considering the block "Fuzzy Inference Loading Manager", in that block there is implemented the algorithm which, when loading the membership functions and the fuzzy inferences, recognizes identical membership functions, storing only one of these for each type and recompiling the fuzzy inferences in such a way that the pointers to identical membership functions converge on the single membership function (of that type) stored.

Further, another block or module "Fuzzy Rule Processing Control Unit"", is to be noted, which in the case the value alpha is computed, implements the search algorithm for an alpha possibly already calculated and therefore stored in the "Support Store" and, in the case none is found, computes it and, in addition to using it for inference, memorizes it in the support store together with the corresponding membership function and the input value that have generated it.

Algorithm for loading the membership functions and fuzzy inferences

As already stated, for computing the fuzzy inference via a structure adapted to compute it, the fuzzy inference (hereinafter called IF) is coded in order to be able to store it within the computing structure, in such a coding one also has that the discourse universe of the membership functions for all the input variables is translated into a basic discourse universe, in such a way to make it possible to compute the fuzzy inference.

By considering a digital computing structure, the basic discourse universe for all the membership functions is mapped onto a discrete set which goes from 0 to(2^n)-1 where n is the number of bits set as the dimension for all the input variables.

After these operations, one has a consequent coding of the fuzzy inference, which is called IF′ and a coding of the membership function which is called M′.

The method for storing has no relevance for our patent proposal since we start from fuzzy inference and membership functions coded with any methods.

A common feature to all the methods for coding is that IF′ must contain the descriptions of the M′ required for computing alphas.

Our patent proposal is the circuit enclosed in the block "Fuzzy Inference Loading Manager", which manages storing IF' and M' within the computing structure.

Specifically, our circuit tells whether a given IF' is pointing to a M' equal to another one already stored and consequently re-directs pointing to this latter by saving memory space.

By considering that a possible embodiment of our circuit is the implementation of a state machine, one can describe in terms of an algorithm one of the several possible implementations by means of a state machine of the hardware circuit included in the patent proposal, specifically the state machine will have to perform the following operations:

A) RECEIVES FROM OUTSIDE IN A SERIAL MANNER THE IF' TO BE STORED

B) STORES THE IF' UP TO THE FIRST CODING M'

C) CHECKS WHETHER THE CODING M' IS PRESENT OR NOT IN THE STORE AND THEREFORE:

C1) IF NOT PRESENT IN THE STORE, STORES IT AND WRITES INTO IF' ITS POINTER

C2) IF PRESENT (BECAUSE IT IS IDENTICAL TO ONE ALREADY STORED), TAKES THE POINTER AT M' ALREADY STORED AND WRITES IT IN IF' (IN THAT WAY THE TWO IDENTICAL MF ARE CONSOLIDATED)

D) RETURNS TO POINT A) UNTIL THE IF's TO BE STORED END.

Algorithm for searching, computing and storing the value alpha

As already stated, quite often the value of the variable V (which is an input to our system) as used for computing alpha, changes with a very low frequency whereby the same value for alpha is computed cyclically a number of times.

Our patent proposal is intended to save time in computing fuzzy inference and therefore in saving resources for devoting them to other processing tasks performed within our computing structure.

Our patent proposal proceeds by determining a value for alpha by searching the value within the support store and, if that is not found, computes the value for alpha and, in addition to making it available for computing the total inference, stores it in the support store.

That support store is used as a stack memory which is loaded from the top and emptied from the bottom. Specifically, in such a memory, the alphas are stored from top to bottom while being computed, and the values found within the memory are caused to move towards the highest point (input points for the new values computed). As the memory is gradually filled, the values stored at the lower point are lost.

By always considering the possible embodiment of the hardware circuit described herein can always be a state machine, such hardware circuit performs the following operations in terms of algorithm:

A) RECEIVES THE PARAMETERS FOR COMPUTING ALPHA, OR THE INPUT VARIABLE V AND THE POINTER TO M' CALLED PM

B) PROCEEDS BY COMPUTING THE VALUE ALPHA AND IN PARALLEL PROCEEDS BY SEARCHING THE POSSIBLE VALUE ALREADY CALCULATED VIA THE VALUES V AND PM.

SPECIFICALLY, IN THE SUPPORT STORE FOR THE VALUES OF ALPHA CALCULATED, THEY ARE STORED {[PM], [ALPHA]} (see figure 2). WITH ALPHA THAT IS THE VALUE CALCULATED FOR THAT VALUE OF V AND THE M' POINTED BY PM, ADDITIONALLY IT IS A TOP-LOADED STACK MEMORY.

THEREFORE THE SEARCH IS AFFECTED BY CONTROLLING THAT THE VALUES V AND PM RECEIVED AT A) ARE IDENTICAL TO THOSE MEMORIZED IN THE FIRST PART OF THE SUPPORT STORE.

C1) IF THEY ARE FOUND TO BE IDENTICAL, THEN COMPUTING THE VALUE FOR ALPHA IS DISCONTINUED SINCE THAT VALUE IS THE ONE WRITTEN IN THE SECOND PART OF THE STORE

LINE AND THE STORE LINE CONTAINING THE VALUE FOUND IS CAUSED TO MOVE TO THE TOP OF THE STORE.

C2) IF NOT FOUND, THEN THE VALUE FOR ALPHA HAS BEEN JUST CALCULATED, ALL THE VALUES STORED IN THE SUPPORT STORE ARE CAUSED TO MOVE DOWN ONE STEP, BY LOSING THE VALUE CONTAINED AT THE BOTTOM OF THE STORE WHILE THE THREE VALUES {[V, PM], [ALPHA]} ARE STORED IN THE LINE AT THE TOP OF THE MEMORY (THAT WAS PREVIOUSLY EMPTIED).

D) ONE PASSES THE VALUE FOR ALPHA TO THE CIRCUIT THAT PROCEEDS WITH COMPUTING THE INFERENCE AND THE PROCESS RETURNS TO POINT A).

Conclusions

In figure 1 a functional block diagram is shown implementing the subject matter of our patent proposal.

Translation of labels in figure 1 of Appendix A

Membership Function Memory

Fuzzy Instruction Memory

Signals

Fuzzy Inference Loading Manager

Unit For Computing Alpha Value

Programming Bus

Signals

Fuzzy Inputs

Fuzzy Rule Processing Control Unit

Support Store

Mux

Computed Value For Alpha

Translation of labels in figure 2 in Appendix A

New values enter

The oldest values exits

SUPPORT STORE

(lettering on right hand side - vertical)

For inserting new values

A value which was found goes to the top